(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
8 November 2001 (08.11.2001)

PCT

(10) International Publication Number
WO 01/84779 A2

(51) International Patent Classification[7]: H04L 12/28, 12/66, 29/06

(21) International Application Number: PCT/CA01/00549

(22) International Filing Date: 23 April 2001 (23.04.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/559,278 27 April 2000 (27.04.2000) US
2,307,521 28 April 2000 (28.04.2000) CA

(71) Applicant (*for all designated States except US*): **SOFT TRACKS ENTERPRISES LTD.** [CA/CA]; Suite 1258, 13351 Commerce Parkway, Richmond, British Columbia V6V 2X7 (CA).

(72) Inventors; and
(75) Inventors/Applicants (*for US only*): **HAWTHORN, John, Bradley** [CA/CA]; 9511 Kirkmon Crescent, Richmond, Bristish Columbia V7E 1M7 (CA). **BELCHER, Richard, D.** [CA/CA]; 245 West 63rd Avenue, Vancouver, British Columbia V5X 2H9 (CA). **VANDENBOSSCHE, Curtis, Allen** [CA/CA]; 8868 184th Street, Surrey, British Columbia V4N 3T3 (CA). **NOBLE, Keith, Edward** [CA/CA]; #207 2170 West 3rd Avenue, Vancouver, British Columbia V6K 1L1 (CA).

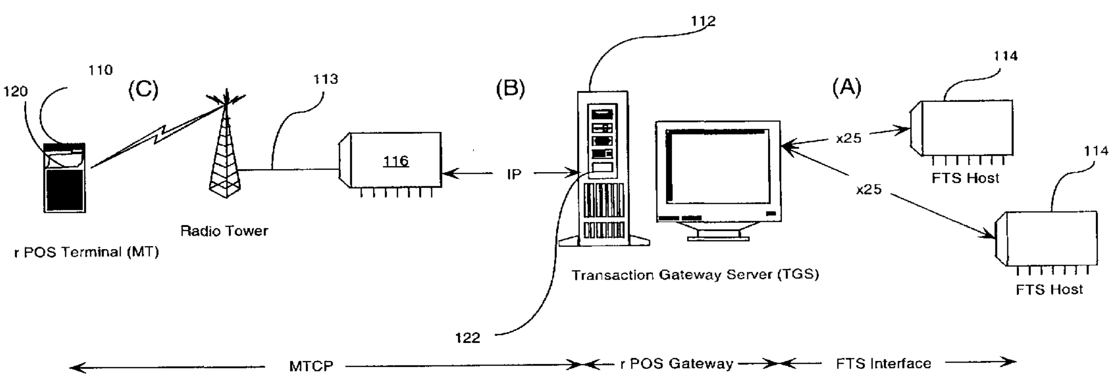
(74) Agents: **PILLAY, Kevin** et al.; Fasken Martineau DuMoulin LLP, Toronto Dominion Tower, Box 20, Suite 4200, Toronto-Dominion Centre, Toronto, Ontario M5K 1N6 (CA).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: REMOTE POINT OF SALE SYSTEM



100

(57) Abstract: A point of sale system comprising at least one point-of-sale terminal for transmitting transaction request in response to a user input; and a gateway server (TGS) for receiving said transaction requests and reformatting the request in conformance with a host financial transaction processing computer connected to the gateway server, such that data traffic between the terminal and the TGS is reduced. In an alternate embodiment, the invention provides a method for communicating information between a terminal and a host in a wireless point of sale system, the method comprising the steps of sending a transaction request to a transaction gateway server; establishing a session between the transaction gateway server and a financial processing host computer; reframing the transaction request at the gateway server for transmission to the financial processing host; sending the reframed request to the host; receiving at the server a response from said host; and generating at said server a terminal response from said host response, whereby the host and terminal do not communicate directly. In a further embodiment of the invention there is provided a method for handling exeption conditions in a transaction between a terminal and a transaction processing host.

WO 01/84779 A2

Published:

— *without international search report and to be republished upon receipt of that report*

REMOTE POINT OF SALE SYSTEM

The present invention relates to a system and method for effecting a wireless electronic transactions and more particularly to a system and method wherein transactions are effected via a transaction gateway server for managing data transmissions between one or more point of sale terminals and a host.

BACKGROUND OF THE INVENTION

In traditional merchant systems, cash registers are connected electronically to an enterprise server. The individual cash registers record payment transactions in all forms (debit card, credit card, cash, or cheque) and transmit transaction information to the enterprise server. Debit and credit transactions are cleared with the merchant's bank and also reported to the enterprise server along with other forms of payment. At the end of the day (or shift), the register transactions are balanced against the enterprise server for all payment transactions, along with any coupons, returns or refunds processed during the day. The total debit and credit transactions are also balanced against the merchant's bank to verify the expected deposits to the merchant's account. The traditional merchant system thus provides a quick and efficient accounting of the day's receipts requiring the least amount of paperwork and labour by the merchant.

While traditional merchant systems require customers to be present at the merchant's premises, a wireless merchant system having mobile terminals allows electronic payment away from the merchant premises creating new business opportunities for the merchant. For example, Internet shopping with "payment-at-the-door" opens new marketing channels with increased sales. Furthermore, established mobile businesss such as courier, taxi, and home services can increase competitiveness and efficiency by offering credit and debit payment to their customers.

A wireless merchant system typically comprises one or more wireless point of sale (POS) terminals connected via a wireless network through a gateway to a financial

transaction server (FTS) which is typically the merchant's bank, often referred to as the acquiring bank.

There are several problems associated with wireless POS Systems. Generally, these wireless POS terminals have no connection to the enterprise server. Instead, they are connected to the acquiring bank, rather than to the merchant's electronic cash register system. Furthermore these POS terminals do not handle cash or cheque transactions. In order for the merchant to close out the day's business, the wireless POS terminals must be brought back to the store along with cash, cheques and individual receipts. Since cash transactions are not recorded, they must be manually entered into the cash register system, opening the door for errors. The debit and credit transactions may be balanced between the POS terminal and the acquiring bank, but there is no record of these transactions on the cash register system so these credit and debit transactions must also be manually entered into the cash register system and/or the merchant's enterprise system.

Due to this need to manually re-enter wireless POS transactions into other systems, balancing transactions between the merchant's bank and the terminal and the cash register system is difficult. Thus these wireless POS terminals, although extremely beneficial for increase service and sales have proven to be too costly to operate in current merchant systems.

Another problem with wireless POS systems relates to the fact that they connect to the transaction-processing host over a wireless network such as a packet switched network. In order for the POS terminal to establish a reliable end-to-end a connection with the transaction-processing host, a large number of packets must be sent over the wireless network. The large number of packets ensures an appropriate level of redundancy and hence reliability in the transaction information. A lost or aborted transaction between the terminal and host also requires reversal information to be eventually sent over the network. This can be costly since the user of the network is normally charged on a per-packet basis. Consequently, various solutions have been

proposed to reduce wireless network usage while ensuring a reasonable level of reliability.

For example U.S. Patent No. 5,907,801 proposes a system for reducing air-time by employing an adapter between the terminal and the wireless modem, that formats all transactions into a common format and compressing the common format data for wireless transmissions.

On the other hand, U.S. Patent No. 6,011,790 describes a POS system wherein the time required to carry out a transaction is minimized, while the burden of data being transmitted over the wireless network is also minimized. This is achieved by establishing a virtual connection between a gateway and the host for each terminal, and giving each terminal control of the end-to-end communication. Circuit management activity is carried out by the gateway in response to bytes received from the terminal, offloading this activity from terminals, thereby reducing on-air traffic. This system does not however provide a solution to the problem of increasing reliability of wireless transactions.

In U.S. Patent No. 6,018,770 a solution is proposed, wherein the local terminals embed connection identification information into each request packet requesting a connection to a remote host system, and the gateway filters this information and verifies that the gateway and local terminal(s) are synchronized. The gateway then manages the communication to the host system using the native protocol of the host, thereby minimizing communication over the packet-switched network. By synchronizing the connection attempts of the local terminals to the gateway, problems associated with connecting multiple transaction-generating terminals to one or more host systems via a packet-switched network are minimized. This synchronization ensures that responses from a particular remote transaction-processing host are passed through to the appropriate POS terminal and that aborted transactions are not authorized.

Thus, there is a need for a method and system which reduces redundant data sent over a network and which also assures reliability of the transaction. There is also a need for such a system to be easily adapted to new POS terminal types and transaction–processing host types while maintaining reliability of the transaction. In other words there is a need for a system that simulates a reliable communication channel, even though in reality the channel is unreliable. Still further there is a need for a system and method that can be easily integrated with a merchant's cash register or back-end enterprise server system while providing similar benefits to the traditional merchant system.

SUMMARY OF THE INVENTION

An advantage of the present invention is to provide a system and method for reducing information transmission across a network.

A further advantage is to increase reliability of transaction integrity between a mobile terminal and a host so as to minimize the number of lost or aborted transactions which result in transaction reversals and thereby reducing out of balance conditions between the terminal and the FTS host, thereby providing a transaction reliability similar to local enterprise merchant system. Although the invention is most applicable to wireless networks, it is also generally applicable to landline networks.

Another advantage is to provide a wireless POS system that provides efficient balancing of various transactions between the remote POS terminals, cash register and the merchant's bank.

In accordance with this invention there is provided a point of sale system comprising:

(a) at least one point-of-sale terminal for transmitting transaction request in response to a user input; and

(b) a gateway server (TGS) for receiving said transaction requests and reformatting the request in conformance with a host financial transaction processing computer connected to the gateway server, such that data traffic between the terminal and the TGS is reduced.

The gateway server and the point-of-sale terminal include software programming that enables each to implement a communication protocol for reducing the data traffic there between, and whereby the server manages the entire session between the terminal and the host.

In accordance with a further aspect of the invention messages transmitted between the terminal and the gateway server are formatted in accordance with the communication protocol. More specifically in a data packet network, the TGS/POS terminals communicate using this protocol so as to minimize the size and number of data packet transmissions over the carrier networks. The protocol makes use of the type and content of data format for elimination of unnecessary static data.

In accordance with a further aspect of the invention there is provided a method for communicating information between a terminal and a host in a wireless point of sale system, the method comprising the steps of:

(a) sending a transaction request to a transaction gateway server;

(b) establishing a session between the transaction gateway server and a financial processing host computer;

(c) reframing the transaction request at the gateway server for transmission to the financial processing host;

(d) sending the reframed request to the host;

(e) receiving at the server a response from said host;

(f) generating at said server a terminal response from said host response, whereby the host and terminal do not communicate directly.

A still further aspect of the invention includes software programming at the terminal and gateway server for monitoring and detecting failures in communication in the network link between the terminal and the host and for generating an appropriate response or exception request in response to the detected failure.

Another aspect of the invention provides for an enterprise reporting system which includes software in the terminal for compiling and forwarding wireless POS transaction information to a merchant enterprise system first over the wireless network and then over a fixed network such as the Internet or a dedicated connection.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other features of the preferred embodiments of the invention will become more apparent in the following detailed description in which reference is made to the appended drawings wherein:

Figure 1 is a general block diagram of the hardware components of a wireless system according to an embodiment of the present invention;

Figure 2 is a block diagram showing information transmission represented in terms of communication layers;

Figure 3 is a schematic block diagram of a transaction gateway server according to an embodiment of the present invention;

Figure 4 is a schematic diagram showing potential failure sections in the network links;

Figure 5 is a ladder diagram showing an exception case; and

Figure 6 is a general block diagram of the hardware components of an enterprise reporting wireless system according to a further embodiment of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

In the following description like numerals refer to like structures in the drawings. For convenience the definitions, acronyms, and abbreviations used in the description are listed in Table I.

Table I

Acronym/ Abbreviation	Expansion
TGS	Transaction Gateway Server
MTCP	Mobile Transaction Communications Protocol
MT	Mobile Terminal
CDPD	Cellular Digital Packet Data
ASCII	American Standard Code for Information Interchange
FTS	Financial Transaction Server, typically operated by the acquiring bank or processor acting on behalf of the acquiring bank.
POS	Point Of Sales
FID	FTS datagram Field Identifier, identifies the various fields within datagrams sent to the FTS (requests) and responses received from the FTS (responses)
DID	FTS Data Download Identifier, identifies the data field that is being downloaded to the terminal
OFID	MTCP Optional Field ID
EFID	ECR Extension Transaction Field ID

Referring to Figure 1, there is shown a block diagram of a wireless point of sale (POS) system 100 according to an embodiment of the present invention. The system 100 preferably includes at least one wireless local point of sale terminal (MT) 110 connected via a transaction gateway server (TGS) 112 to at least one financial transaction server (FTS) or host computer 114. The POS terminal 110 communicates via a wireless network 113 to the TGS 112. In a preferred embodiment the wireless network is a packet data network 116 which in turn communicates directly with the

TGS via an Internet protocol (IP) connection. In a preferred embodiment the TGS communicates with the host 114 using an X.25 protocol or similar protocol. The wireless network may be a GPRS network, a CDPD(cellular digital packet data) network or such like.

Although the above describes the system components of a typical wireless transaction system, the system of the present invention provides advantages over current systems as will be described in detail below.

Firstly, the system of the present invention replaces the traditional gateway-only system with a gateway server where the wireless terminals do not communicate directly to the financial host nor does it require specific knowledge of the host communication requirements. Instead, the transaction gateway server of the present invention performs the host communication functions and retains some of the terminal **specific static** data. Using this model, the terminal can minimize the data sent per transaction by sending only the captured data. Terminal static data is sent only once during log on to the TGS, stored in the TGS and used to build an FTS host compliant transaction request message when the terminal sends a transaction request to the particular FTS host.

Accordingly, the present invention provides application software on the terminal and application software on the server for implementing a novel communication protocol between the server and the terminal. The novel communication protocol according to the present invention allows separation of the host functionality from the terminal. Thus allowing errors or breaks in communication to be handled more efficiently than current gateway only systems.

The following is a more detailed description of the system 100 components depicted in figure 1. The POS terminal device 110 is preferably a fully integrated point of sale unit with PIN/POS pad, card swipe, display, printer and battery power system. The POS device also includes an integrated CDPD wireless network wireless RF modem (not shown). In one embodiment, the hardware platform selected for the POS is the

IVI Checkmate "Elite 780 RF CDPD" device, but could alternatively be any other type of point of sale transaction processing machine such as a personal computer, a laptop computer, smart phone or stand alone vending terminal.

The terminal application software 120 resides in the POS device. The terminal application layer software 120 according to one embodiment of the present invention enables the POS terminal 110 to communicate with the TGS 112. The TGS 112 includes software 122 for performing host communication functions and for retaining terminal specific data, the implementation of the terminal and POS application software will be described more fully below.

The terminal application software 120 performs the tasks of collecting sales information details as required to complete a sale and in the manner required by the financial institution prompt syntax; building MTCP datagrams needed by the TGS to reconstruct base 24 requests for transmission to the FTS, this includes ensuring data integrity by passing encrypted data directly to the FTS without modification. Furthermore the application program maintains a session with the TGS and utilizes the novel communication protocol herein termed a "Mobile Transaction Communication Protocol" (MTCP) (to be described below) to ensure that the data arrives at the TGS or if not, then to provide appropriate alternative action or exception handling. The MTCP protocol, is adaptable to accommodate the various versions of financial institution data, end to end, while at the same time placing a minimal burden on the service features of the wireless data network thus providing a network independent of communications protocol. The application program also detects and corrects network layer exception conditions such as loss of contact with the TGS. Finally, the terminal application program interfaces to the RF data network modem.

The TGS 112 is a computer that attaches two or more networks and routes packets between them. Thus the TGS is also a gateway that interfaces with FTS processors and wireless networks. Similar to the mobile terminal, the TGS communicates using the MTCP protocol (to be described below) over the wireless networks and uses base24 messaging to communicate with the FTS host or similar financial transactions

protocol's like ISO 8583. It builds complete base24 messages for transmission to the FTS, and this ensures data integrity by passing encrypted data untouched to the FTS host. It also maintains a session with the terminals and utilizes the MTCP protocol to ensure that data arrives at the terminal or that the appropriate action is taken if data does not arrive. It further detects and corrects network layer exception conditions (i.e., loss of contact with the FTS host). Furthermore, the TGS interfaces to the financial institutions (typically, link layer framing over X25) and interfaces to the various wireless networks such as IP, X25 and other proprietary network layers.

The system 100 may also include an enterprise reporting subsystem (ERS) which includes reporting software in the POS terminals 110 and a bank open exchange (BOX) server transaction which is connected via the wireless data network to the POS terminals. The BOX server compiles and forwards the wireless POS transaction information to the merchant's enterprise system 120 over the Internet or a dedicated connection. The BOX server acts as a data warehouse. The BOX server together with its enterprise reporting and outputs way people gain access to the data and the various input and output methods all together, comprise the Enterprise Reporting Subsystem or ERS. This aspect of the invention will be described in greater detail with reference to figure 6.

The system 100 may also be described in terms of three basic communication interfaces or communication channels namely, the TGS to FTS host channel (A), the TGS to the wireless network (B), and the POS terminal to the wireless network interface (C).

The TGS to FTS host interface (A) is a well-defined and standardized interface. The reader is referred to US Patent No. 6,018,770 for a description of such an interface. This interface is typically based on the base24 credit and debit message formats transported to the FTS hosts over an X.25 circuit (described in reference standards). Although based on a standard other than variations of base 24, variations of ISO 8583 are also often used for the communication interface (A).

Accordingly one aspect of the present invention provides for a uniform standard host or device interface and standardized services for processing wireless data in the interface between the terminal 110 and the TGS 112. In a preferred embodiment, CDPD networking is implemented and thus the interfaces (B) and (C) conforms to "CDPD System Specification Release 1.1". In addition, the interface between the point of sale terminals 110 and the TGS implement the new communication protocol (MTCP).

Referring now to figure 2, the message in a typical transaction of the present invention is shown schematically by referring to an OSI protocol stack diagram showing communication between the various layers at various points in the system. The stack diagram represents a message (indicated by the shaded square) flow from the terminal throught the TGS to the FTS using a typical CDPD network. For each link in the system the corresponding communication stack is illustrated. The message block at the bottom of each stack is a schematic representation of a complete datagram. As with the OSI 7 layer model, it is understood that each layer communicate with its peer layer at the other end of the communication link.

The terminal application software builds a stack as depicted in figure 2 as Stack-A. The POS terminal interfaces to the CDPD network using an IP network layer and SLIP link layer to the RF modem. The terminal MTCP message is encapsulate in the IP and UD datagrams. The message is then passed to the wireless network.
The TGS receives the MTCP message and using stored information forms a complete SPDH message. The TGS applies link layer framing and forwards the message to the TCP-to-X.25 gateway. This is indicated by Stack-C. At this point MTCP is no longer part of the message, rather standard SPDH messaging is now used.

Thus MTCP type messages flow between the terminal and the TGS and SPDH messages flow between the TGS and the FTS host. Regardless of communication stacks, some types of information must flow end to end from the terminal to the FTS host (or vice versa). This information that must flow end-to-end is generally information that is unique for each transaction, or information that must not be altered

(such as encrypted data) for security integrity. As mentioned above a the shaded square is used in figure 2 to represent this end-to-end data.

A second type of information or data is generally static, or seldom changing data (perhaps only when the terminal logs on or off). This type of information can be exchanged between the TGS and the terminal once at session logon, thereafter, the TGS can store the information and use it as needed to construct complete SPDH messages. This information is depicted as a dark black square in figure 2. It is to be noted that this type of information only flow between the TGS and the FTS host.

MTCP consists of two main components namely, Transport (record format, error handling); and Application (specific transaction formats for requests and responses and exception handling). This is more clearly shown in figure 2.

The Transport defines message format, error handling, message **passing** and, rules for large records. The Application defines the way a record is built, the field requirements to complete a specific transaction for a specific host and the response to the transaction request.

In wireless communications normally all communication devices stay in a passive listen mode. However for portable terminals this is not always practical since this results in a continual power draw from the terminal power supply, typically rechargeable batteries. In order to preserve power and to maximize the operational time of the terminal the radio (and other power consuming devices) is turned off when it is not listening for the response to its transaction request.

In a POS environment this is acceptable since transactions occur sporadically through the course of the business day. Typically an FTS does not send unsolicited messages to a terminal, however MTCP does include support for dispatch type messages (piggyback messages).
The terminal typically turns on the radio and sends it's transaction request to the TGS. It then keeps the radio receiver on and listens for it's response message. When it

receives the response it can continue on with it's process. However, if a response does not arrive and the response timeout period expires, the terminal proceeds with the appropriate error handling.

The following process outlines the communication between terminal and TGS:

(a) No activity occurs between terminal and TGS until an operator physically performs an action at the terminal;

(b) Communication is always initiated by the terminal as a result of some action from the terminal operator;

(c) The terminal forwards a transaction request to the TGS via the wireless packet network.

(d) The TGS receives the terminal request, retrieves additional data, re-frames the request and sends it to the appropriate FTS host;

(e) When the response from the host is received, the TGS identifies the original request, does all required calculations, creates the MTCP response for the terminal and sends it to the terminal;

(f) Since all communications are initiated by the terminal, it cannot receive unsolicited downloads from the FTS or TGS. If a download is required (host initiated) a trigger will be sent to the terminal as part of the response. This procedure will be an FTS specific operation and should be detailed in the FTS implementation specification;

(g) Each transaction type may have specific exceptions handling associated with particular transaction type or subtype. The exception handling typically requires a reversal of the failed transaction which may involve the automatic generation of a VOID transaction or a reversal transaction. Reversal methods are dependent on the specific processor and in some instances not permitted by the processor. Exception handling will be described in more detail in conjunction with figure 4.

The MTCP is developed around the concept of sessions. A session is normally established and maintained between the TGS and the FTS and a typical session consists of:

(a) configuration

(b) logon

(c) initialization (depending on requirements of particular hosts)

(d) transaction processing (note: In this system, terminals do not have to establish a session with the TGS/FTS for every transaction)

(e) logoff (although some terminals may be perpetually on)

A session is between the terminal and the TGS to a specific host. Part of the log-off process for the FTS hosts may require batch settlements to be performed. This ensures the terminal balances to be in sync with the specific host. A number of batch operations are performed within a session. For example a session may be established and maintained over a number of user or operator shifts.

Concurrent host support is implemented through the MTCP transport layer.

Terminals are independent from host communication, that is the formatting of host records and maintenance of the host link is not required of the terminals. MTCP provides a level of isolation of the host link, however transactions may require specific knowledge of host exception handling in order to provide support for transaction reversals. The transaction reversal support is dependent on the specific host interface.

The transport layer provides for the transmission of application messages, version identification, peer-to-peer synchronization, large messages and error handling.

Application messages include transaction requests, responses, downloads and other messages that occur between the terminal and TGS. The MTCP format prefixes five bytes to each application message that is used to identify the message.

Version control allows the TGS and terminal to support different MTCP transports, identifies FTS host types, as well as application versions.

Synchronization occurs when the terminal and TGS agree on the parameters for subsequent messages. . Maximum message sizes are restricted for a number of different reasons. Terminal buffers, radio buffers, RF transmission all have limitations that could affect the maximum message length. These need to be adjusted dynamically between the TGS and terminal for each session. This is accomplished using the session parameter message.

In situations where the application message exceeds message size limitations of the terminal or transmission media, the message must be broken up into smaller fragments before sending. MTCP allows the sending of fragments and defines the rules for rebuilding the message at the receiver end as well as assuring the reception of all fragments.

MTCP supports large message transfer to the terminal. A large message may be a file or a transaction message that contains a number of download parameters or response messages. The current limitations on message sizes are terminal related. There is a maximum file size the terminal can support and the maximum transmission buffer size (fragment size) the terminal can support. The maximum file size limits the size for download files. The maximum buffer size limits the size of each fragment that can be received.
The message limits are defined by the parameters: fragment size, number of fragments per message and window size.

Due to transmission methods fragments could be sent and received in asynchronous fashion, that is the fragments can be received not necessarily in the same order as they were sent. This could result in lost fragments which requires retransmission of the particular fragment. The retransmission mechanism is built into the response from the receiver to indicate which fragments are missing. It is up to the sender to determine the fragments not received and re-transmit.

Each acknowledgement also identifies to the receiver if there are more fragments pending. The 'more' bit is used to control the receiver response to the multi-fragment message in rebuilding the larger message.

The TGS only sends responses to a terminal request and does not expect positive acknowledgements from the terminal. This means that the last group of fragments will remain unacknowledged, since the TGS will not receive the last ack from the terminal if all fragments are received properly. The next transmission from the terminal can be any message including a negative acknowledgment from the previous fragment transmission, the TGS must store the last group of fragments for retransmission if required. MTCP uses negative acknowledgments with persistent reversals.

Each MTCP message consists of a header followed by message data. The format of the message is:

Header			Data
Transport Layer Version 2 byte	Sequence No. 2 byte	Control Byte 1 byte	Application Layer Data 1 - X

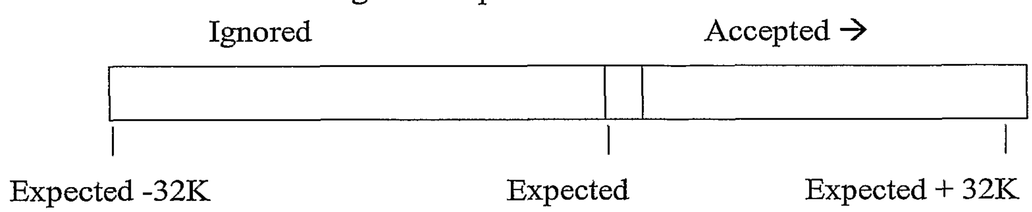
The Version Id is interpreted by the TGS and this identifies the version of MTCP the terminal is communicating with, the application version of the message and the host destination.

A Sequence Number is Used to match responses to requests, and identifies all of the fragments in a multi-fragment message. Peers use a sliding-window mechanism to detect old and new sequence numbers as follows:

If a message arrives within a 32K backward window of the expected sequence number, it is a late message and is ignored.

- If a message arrives equal to the expected sequence number, it is accepted.

- If a message arrives within a 32K forward window of the expected sequence number, then zero or more messages were late/lost in the network. The message is accepted.



A Control byte Supplies transport layer information

There are four basic message types in MTCP:

Normal Message

Fragment Message Acknowledgement

Session Parameters

Session Parameters Error Message

The Message format – Error Response (session parameters error) is sent by a peer in two cases:

A) In response to a Session Parameters request, when the parameters supplied are either out of range or cannot be supported by the receiver.

B) When a message other than Session Parameters is sent to a peer, but the session has not been initialized with it.

Field	Length	Description
Version	2	
Sequence number	2	
Control byte	1	

A Typical Session Flow is shown in the following table:

Sender (Terminal)	Direction	Receiver (TGS)
Sender initializes session by sending Session Parameters message.	→	
	←	Receiver responds with same, supplying its own parameters.
Single fragment application-layer message is sent by a peer.	→	
	←	Receiver has acquired all fragments in this message, and passes it up to application layer.

Long message:

Sender	Direction	Receiver
Sender initializes session by sending Session Parameters message.	→	
	←	Receiver responds with same, supplying its own parameters.
Send of multiple fragment application-layer message is initiated.	→	
Sender gathers next <WindowSize> fragments from message and sends them all at once.	→	
		Receiver acquires several fragments of message.
		If all fragments in this window have been received, and more bit is not set on last packet goto 10.
	←	Receiver sends Acknowledgement, filling in Fragment Bitmap with list of fragments received.
Sender examines bitmap. If all fragments were not received, sender re-sends the missing fragments.		
		All fragments in window were received,.

Data that is sent in MTCP may be represented differently in order to optimize the transmission packet. The basic data types supported within MTCP:

TYPE	Indicator	Lengths	Values
Integer/Signed Integer	I or SI	1 byte	- 127 + 128
		2 byte	- 32767 + 32768
		4 byte	- 2147483647 + 2147483648
Unsigned Integer	UI	1 byte	256
		2 byte	65536
		4 byte	4294967296
Alphanumeric	AN	Any length	ASCII set

Data sent to the FTS host by the TGS are typically Numeric, Financial or Text/Alphanumeric. These data types may be represented in MTCP by the following:

Numeric data - If valid possible values are within the maximum value of Unsigned Integer then UI may be assigned, else text may be required.

Financial/Amounts - Will typically be represented either as I or UI depending on the requirement for negative values.

Text/Alphanumeric - due to various requirements of text fields these fields are not converted.

The standard message format is:

Message Type Code 1 byte	Data message

The format of the data message is determined by the message type code. Request transactions and response messages are comprised only of optional fields. This allows the protocol to support a number of different hosts with little or no change to the record layout. Each optional data field is prefixed with a length byte followed by a unique field identifier (FID).

Optional field format is:

Data Length (1-256) 1 byte	FID 1 byte	Data (1 - 256)

Administration messages, ERS and other messages are formatted with a combination of fixed fields followed optional fields.

There are five transaction groups that are supported by MTCP. These are Log-On (Log-Off), Financial, Administrative, Batch Operations and ERS Extensions. Within each group are transactions that perform specific operations with either the TGS or the FTS.

Before a terminal can be used for transactions there may some initialization that is required at the terminal to ensure it has the proper setup parameters. This is done through log on procedures. There are two levels of log on:

TGS log on

FTS log on (optional, not supported by all hosts)

TGS log on is a local logon. In response the TGS has the option to send a Terminal Profile Download Command to the terminal. Terminal profile download is configuration data that is specific for the terminal to ensure the proper operation and functionality for a particular system/merchant. Profile download may include communication time-outs, buffer length definitions, routing information, additional

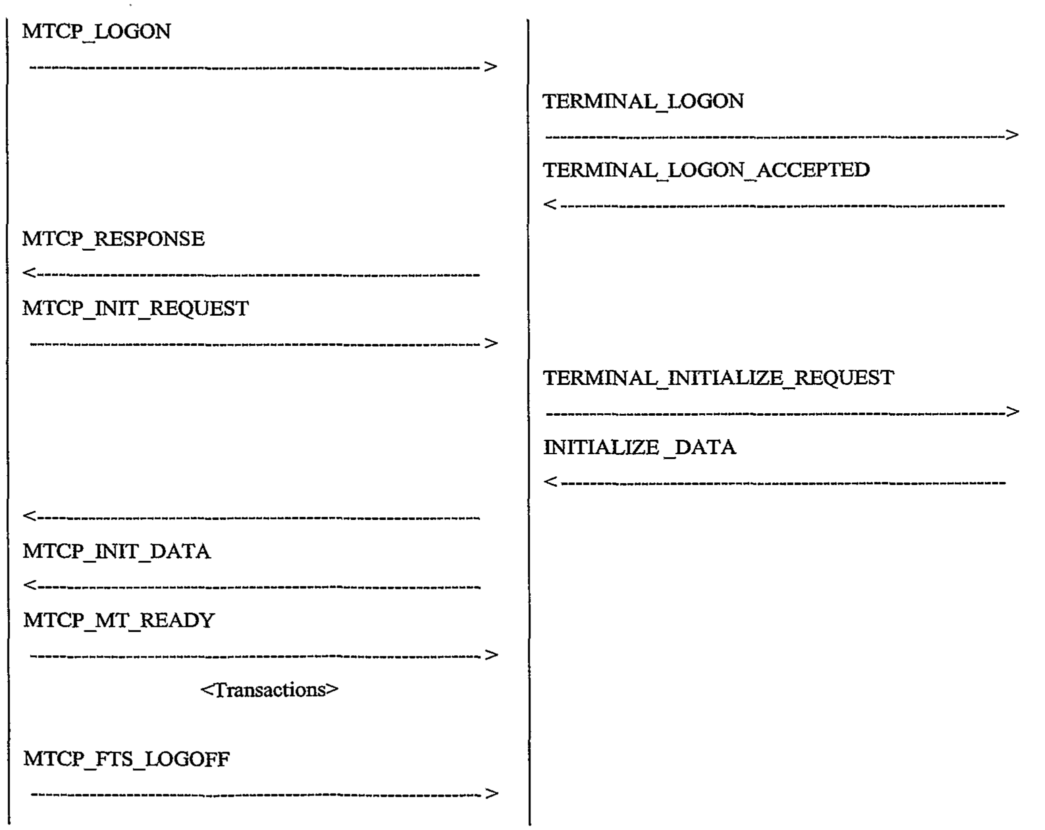
scripting to support new transactions, local terminal (non-FTS related) administration codes, etc.

Typically a terminal profile will seldom change for a particular merchant. However, changes may occur if there are updated system requirements or new merchant needs. In Profile Download, Profile information is passed to the terminal as part of the initialization process. It is performed before logging onto the TGS and initializing the communication to the FTS host. There are several reasons for this:

Profile information must be able to restrict processing of certain transactions based upon the host type, card type and market type.

Global fields such as the application and the terminal type and communication parameters must be identified at this time and configured by the user if required, so that they can be passed back to the TGS during logon and stored for subsequent use.

Terminal **TGS** **Host**



TERMINAL_LOGOFF

-->

TERMINAL_LOGOFF_ACCEPT

<--

MTCP_RESPONSE

<--

The Configuration/Logon/Initialization Process

By doing this, the terminal loads a default configuration for the desired host. Upon Initialization some of this data can be overwritten with parameters originating from the FTS host. The profile information controls which fields can be overwritten at initialization time.

Before requesting configuration information, the MT user must provide the following information:

- FTS Host
- Operator Language ID
- Default Customer Language ID

The Initialization Download Functions performed at FTS log on include connecting to the FTS host and waiting for any Initialization Download from the host which will configure the terminal as expected or defined with the particular FTS host. Initialization download is profile information specific to the transactions with the particular host. Examples of initialization parameters are transaction allowed/disallowed, language, merchant codes, etc. This is defined for each host and requires specific terminal development to support the various parameters available from the particular host.

Initialization is performed after the TGS has logged onto the host. This step has been provided in order to allow hosts to override certain profile information on the mobile terminals on an “as needed” basis and allow some exchange of configuration information between users and the TGS. As an example of this, some FTS systems supply the batch number that the terminal is to used for the first batch following log-on.

For each credit card transaction, the terminal builds and sends a credit request message to the TGS, which in turn builds a request to the host in the specified format and sends this message to the FTS host. Minimum information includes terminal identification, type of transaction, credit card information, invoice number and sale amount. Upon receipt of the credit request the FTS host validates the message contents and sends a response to the TGS indicating approval, referral or denial of the transaction. The terminal is required to display the reason for denial/referral, or approval code for approval.

The general format of a response message is as follows:

Field	Offset	Size	Type	Comments
Message Code	3	1	I	MTCP_RESPONSE
Error Code	4	1	I	Specific to the type of request the message applies to and to the FTS host in use.
Config Flag	5	1	I	Indicates whether the MT needs to reconfigure itself. 0 - No, 1 - Yes.
Message Text	6	1-256	AN	OFID D Variable text. For a description of this field see **Error! Reference source not found.**.
Generic Fields		1-256	AN	OFID Z Variable text.

Message codes identify to the receiver the type of message that the sending unit has transmitted. The message code is know on both ends the transaction that is to be performed by the receiving unit, the data fields that are associated with it as well as the associated exceptions handling routines that may apply to the specific message. If the receiving unit receives a message code that it cannot interpret, no response will be given.

Message codes are single byte value from 1-255. Values 1-127 are reserved for terminal requests while the higher values 128-255 are reserved for TGS responses.

Some messages may also contain a response code from the host in response to the terminal request. Response codes may have also have text message associated that is to be displayed to the operator. Most host specifications have pre-defined host error messages and a fixed text string.

To reduce the amount of data transmission, text messages are tabled in the terminal and only the response code normally needs to be sent. The TGS has the same response code table and each response received by from the host with a text message attached is reviewed by the TGS. If the text message associated with the response code is different from the one expected by the TGS, the entire text message is sent to the terminal. The terminal overrides internal table messages with the received text message.

The TGS maintains error tables per terminal. These tables are reset to the administrator-defined default value when the terminal requests a new configuration.

In addition to this common structure the responses may have additional fields which are described on a message-by-message basis.

As terminals typically can not listen for unsolicited messages, notifications for changes of configuration are sent as part of the response messages. The TGS will set this field until the terminal contains the most recent configuration. It is up to the terminal to reconfigure itself.

Referring to figure 3, there is shown a graphical representation of the TGS architecture. The TGS includes a network layer interface for connection with an RF network, a configuration module which provides access to the configurable settings of the server, a job list module for maintaining a list of currently in-progress transaction. Jobs are created when a new request is received from a terminal and saved here for later retrieval. The TGS also includes a database for storing persistent logon session data and detailed information about financial transactions that have been performed; a custodian for periodically performing maintenance procedures; and a dispatcher for receiving incoming requests from the RF network. When such an event occurs a worker thread is taken from a thread cache, given the request to process and started.

Referring now to figure 4, there is shown the failure point in the various links or paths of a wireless transaction. Path 1 is defined as the CDPD uplink from the terminal to the TGS. Path 2 is defined as the uplink from the TGS to the FTS. Path 3 is the downlink from the FTS to TGS, and Path 4 is the CDPD downlink from the TGS to the terminal. Note that the word "uplink is often replaced with "reverse channel" and the word "downlink" is often replaced with "forward channel".

Referring to figure 5, there is shown a ladder diagram showing a sequence of messages for a failed purchase transaction. The table below describes the response of each of the system components, the terminal or TGS if one of the links fails, during a purchase transaction. For example the ladder diagram of Figure 5 shows the sequence of actions or responses provided by the TGS when the CDPD downlink (4) fails.

	1	2	3	4
Transaction Type	**CDPD uplink**	**TGS to FTS uplink**	**FTS to TGS downlink**	**CDPD downlink**
Purchase	Terminal detects "out of range" (comm 401) and immediately Fails A "CANCELLED" receipt is printed. Transaction is in the failed transaction list.	TGS reports General Host Error to the terminal immediately. The transaction is cancelled and terminal returns to SALES screen. The terminal sends a MTCP debit reversal request. The TGS response "original request never performed"	TGS does not know FTS response. If a MTCP retries arrives before global timer expiry, the TGS will response "STILLPROC" The TGS global host response timer expires and the TGS send a General Host Error to the terminal. A "CANCELLED" receipt is printed.	TGS knows the FTS response. The terminal times out and does a MTCP re-send. If response received, no issues. Terminal eventually exhaust retries and timeout. Record the transaction in the failed transaction list. Terminal sends a MTCP reversal.

		A "CANCELLED" receipt is printed. Transaction is record in the Failed transaction list.	The terminal immediately sends a MTCP debit reversal to the TGS. The TGS nak's the terminals debit reversal request. The terminal must maintain SPDH sequence numbers. Must rely on bank for internal reversal. Record the transaction in the failed transaction list.	request. TGS sends FC(controller) reversal to bank using the known response. The TGS sends MTCP reversal ACK to the terminal. The terminal increments sequence number. OR The terminal eventually exhausted reversal attempts and prompts the user to suspend sending reversals. If the user further uses admin-46 to CANCEL reversal then the transaction is marked with t '*' in the failed transaction list.

In summary, it may be seen that the present invention provides a gateway server that attempts to emulate the host responses to the terminal, such that the terminal only need communicate with the TGS and thus need not be in control of the end-to-end transaction as in prior art systems.

Although the invention has been described with reference to certain specific embodiments, various modifications thereof will be apparent to those skilled in the art without departing from the spirit and scope of the invention as outlined in the claims appended hereto.

THE EMBODIMENTS OF THE INVENTION IN WHICH AN EXCLUSIVE PROPERTY OR PRIVILEGE IS CLAIMED ARE DEFINED AS FOLLOWS:

1. A point of sale system comprising:

(a) at least one point-of-sale terminal for transmitting transaction request in response to a user input; and

(b) a gateway server (TGS) for receiving said transaction requests and reformatting the request in conformance with a host financial transaction processing computer connected to the gateway server, such that data traffic between the terminal and the TGS is reduced.

2. A method for communicating information between a terminal and a host in a wireless point of sale system, the method comprising the steps of:

(a) sending a transaction request to a transaction gateway server;

(b) establishing a session between the transaction gateway server and a financial processing host computer;

(c) reframing the transaction request at the gateway server for transmission to the financial processing host;

(d) sending the reframed request to the host;

(e) receiving at the server a response from said host; and

(f) generating at said server a terminal response from said host response, whereby the host and terminal do not communicate directly.

3. In a point of sale system, a method for handling exception conditions in a transaction between a terminal and a transaction processing host, said transaction being managed by a transaction gateway server, said method comprising the steps of:

(a) transmitting from said terminal a predetermined transaction message;

(b) monitoring at said terminal for a response from said transaction gateway server;

(c) starting a timeout period while waiting for a response from said gateway server;

(d) resending said transaction message during said timeout period;

(e) transmitting a transaction reversal following said timeout period; and

(f) said transaction gateway server determining from said transaction reversal a corresponding reversal expected by said transaction processing host.

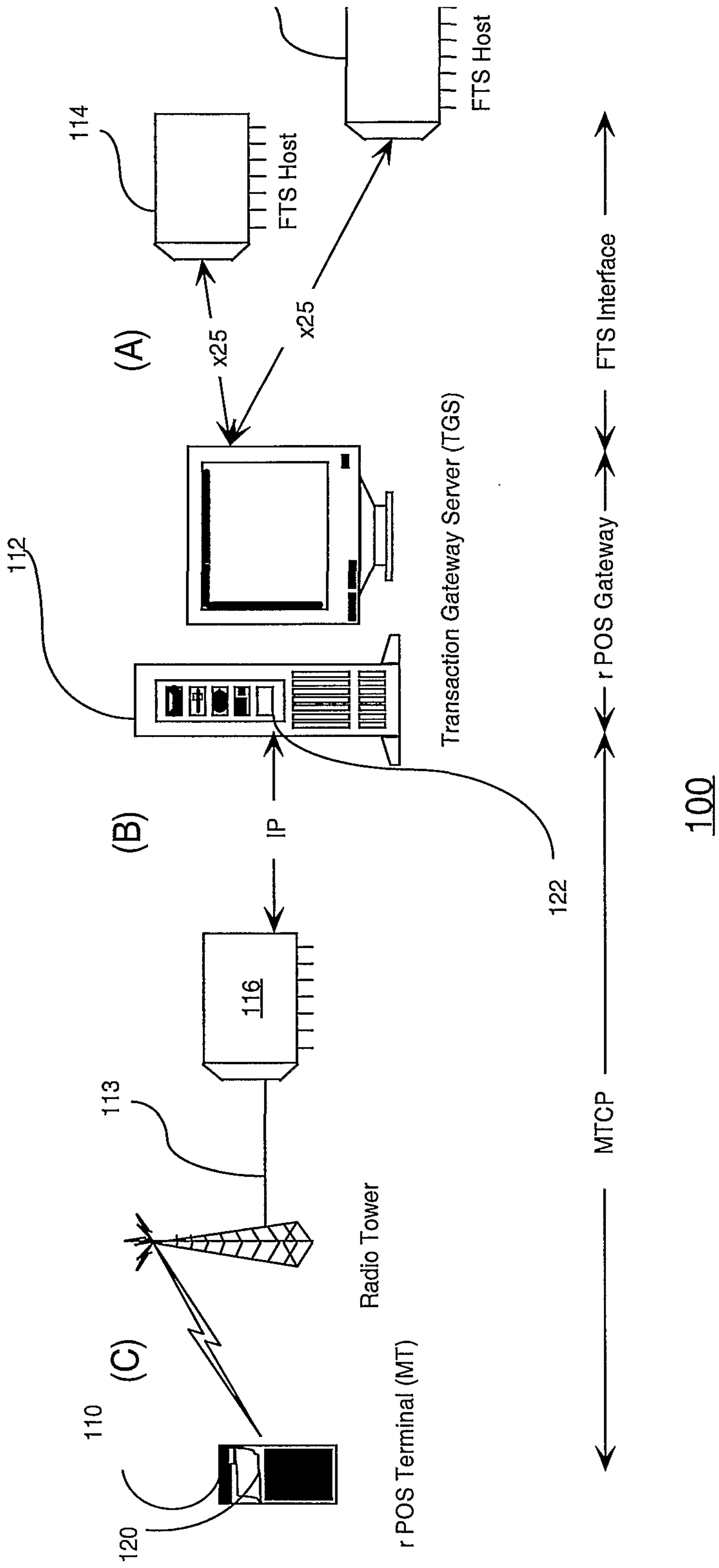

114
FTS Host
FTS Host
(A)
x25
x25
112
Transaction Gateway Server (TGS)
(B)
IP
122
116
113
Radio Tower
(C)
110
120
r POS Terminal (MT)
MTCP
r POS Gateway
FTS Interface
100

Figure 1

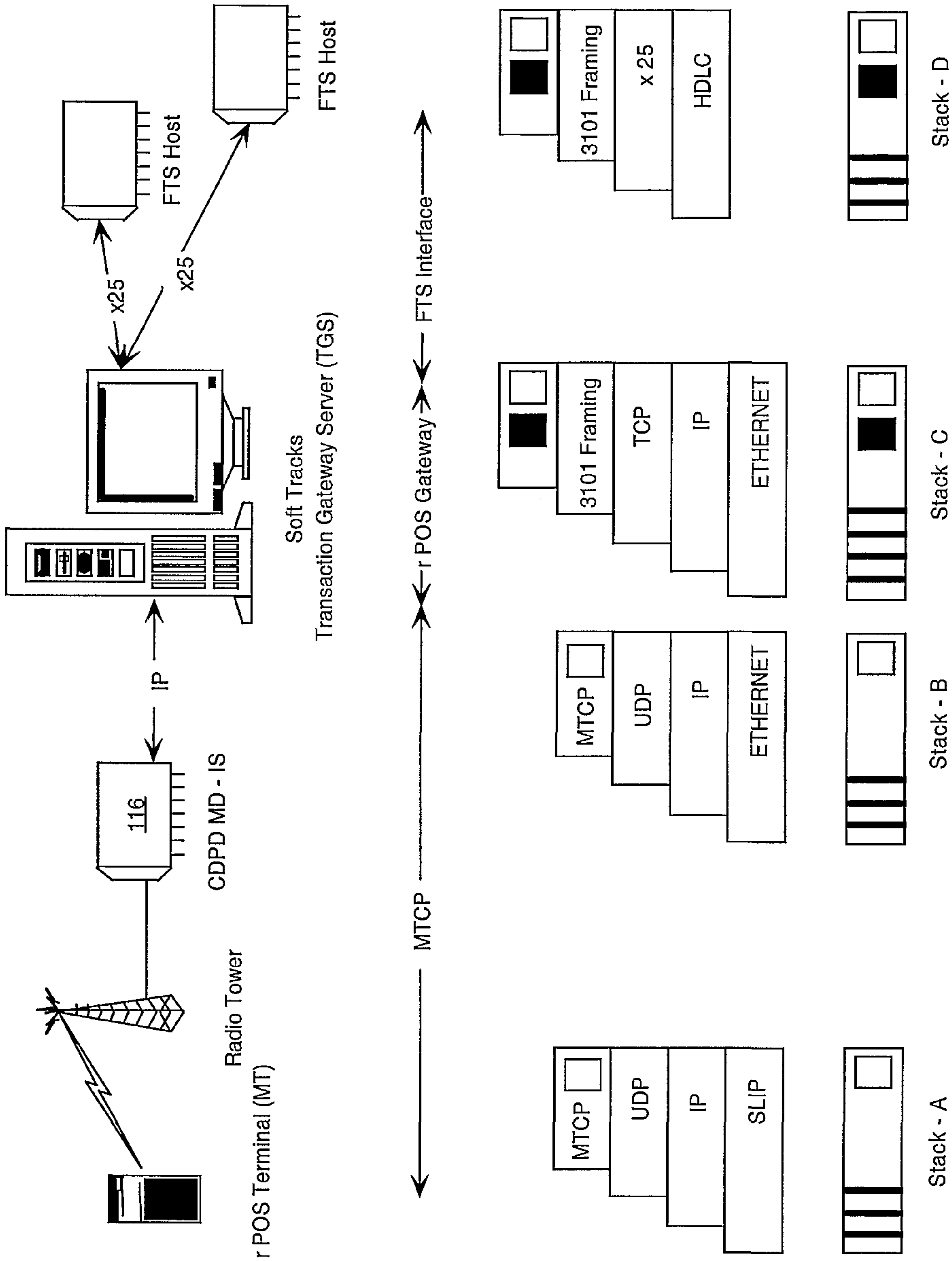

r POS Terminal (MT)
Radio Tower
116
CDPD MD - IS
IP
Soft Tracks
Transaction Gateway Server (TGS)
x25
x25
FTS Host
FTS Host
MTCP
r POS Gateway
FTS Interface
MTCP
UDP
IP
SLIP
Stack - A
MTCP
UDP
IP
ETHERNET
Stack - B
3101 Framing
TCP
IP
ETHERNET
Stack - C
3101 Framing
x 25
HDLC
Stack - D

Figure 2

Graphical Representation

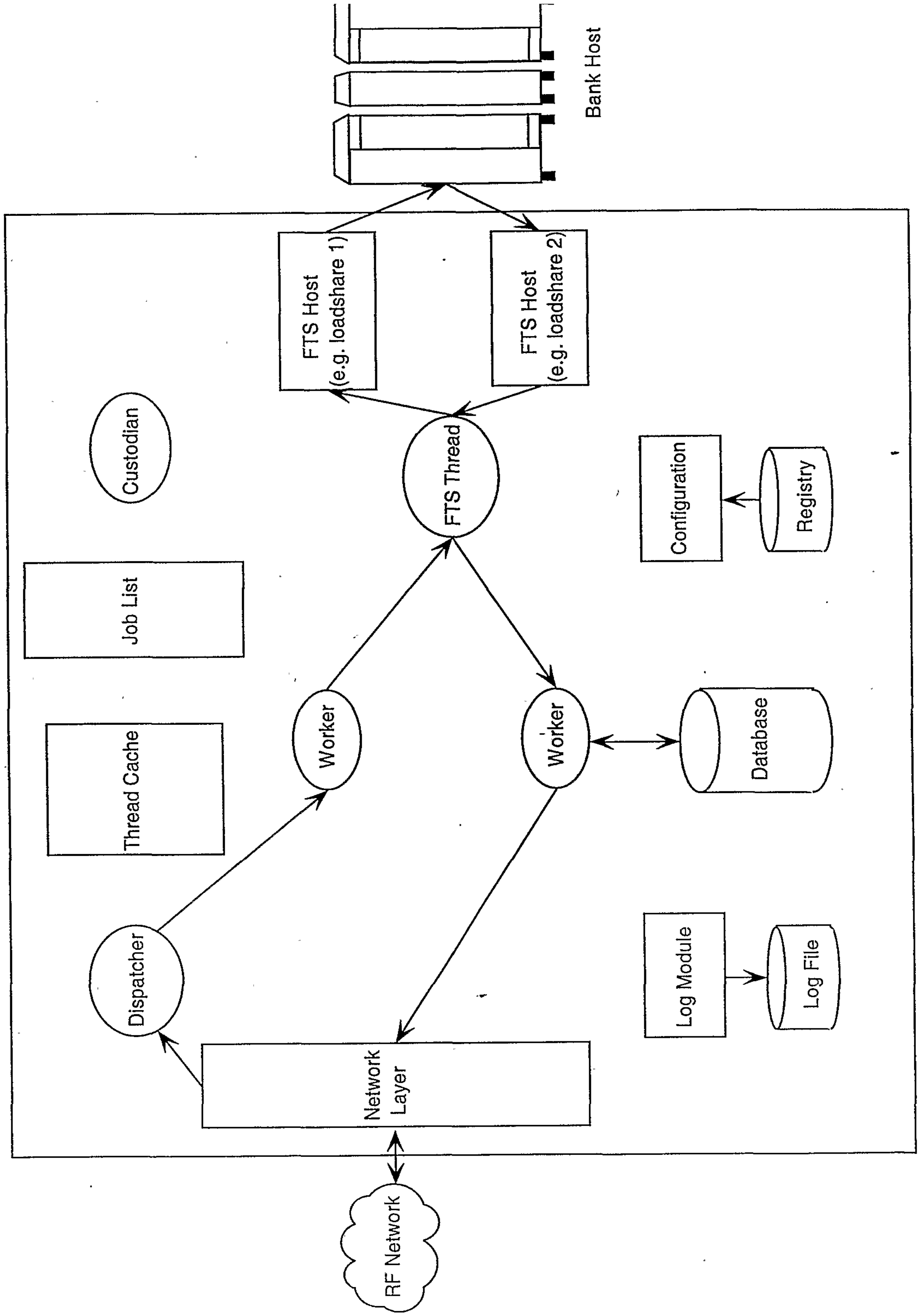


Figure 3

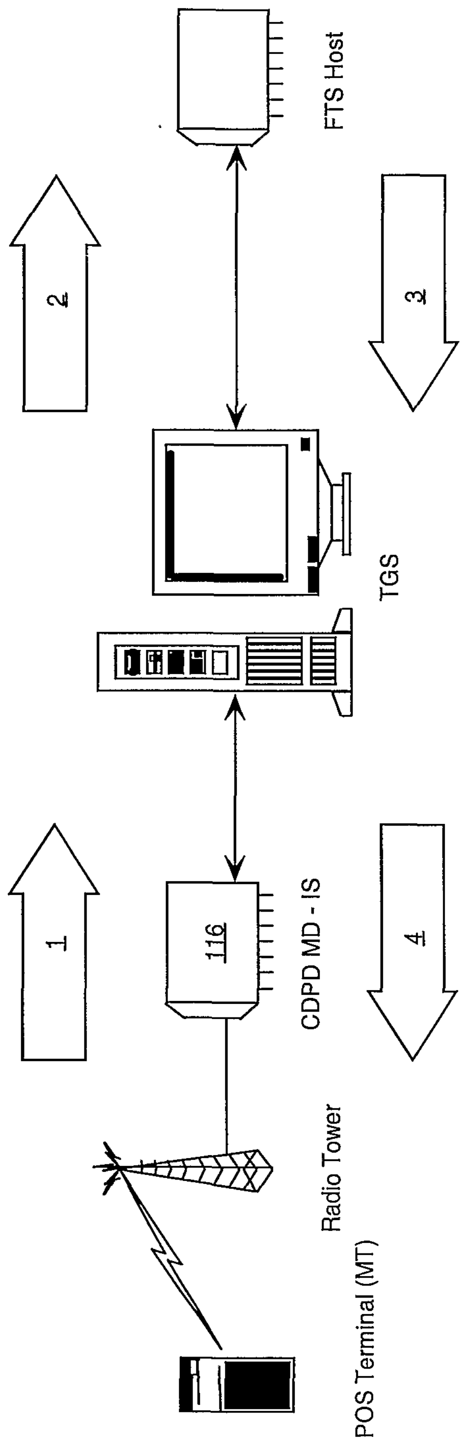


Figure 4

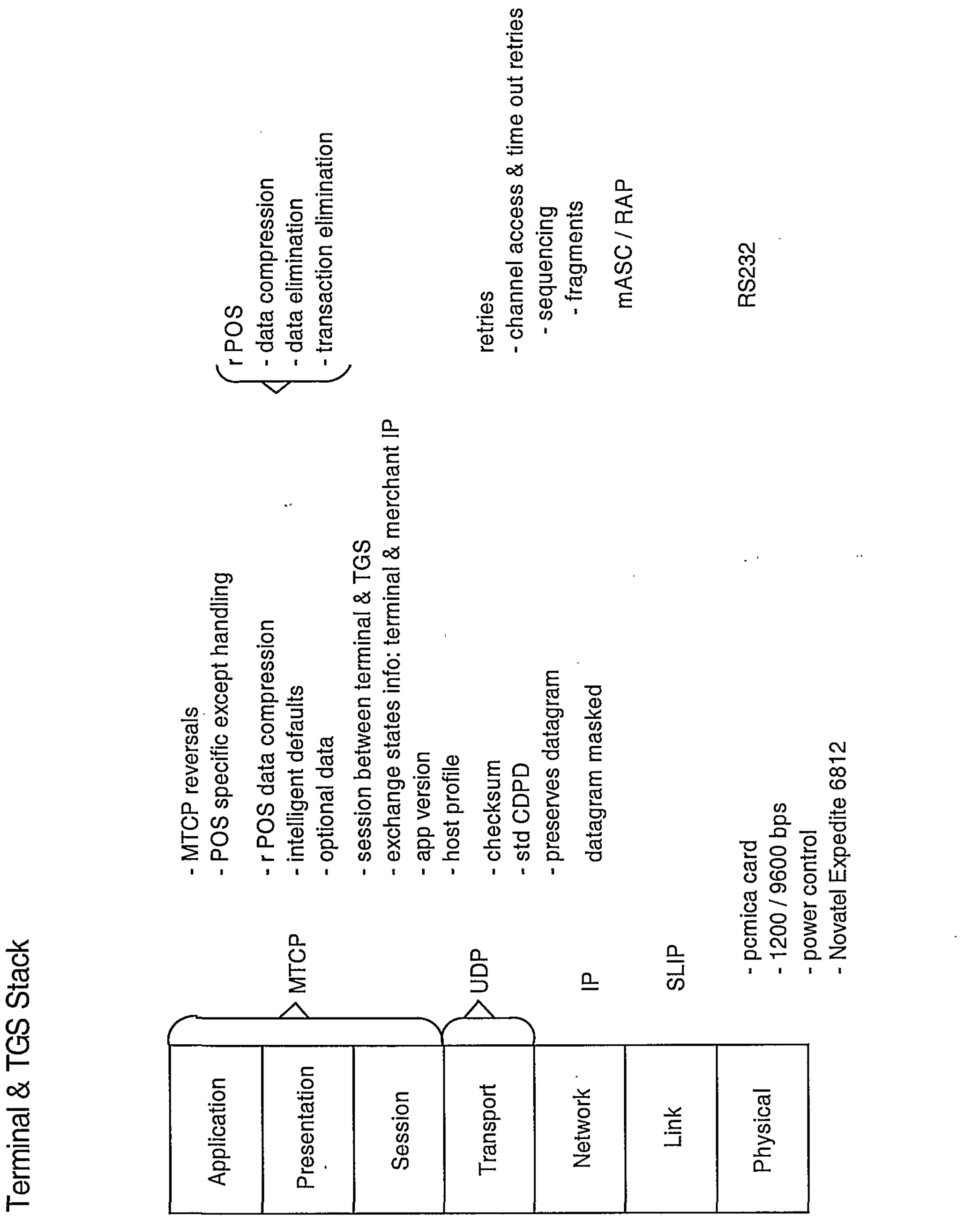


Figure 4 (a)

SUBSTITUTE SHEET (RULE 26)

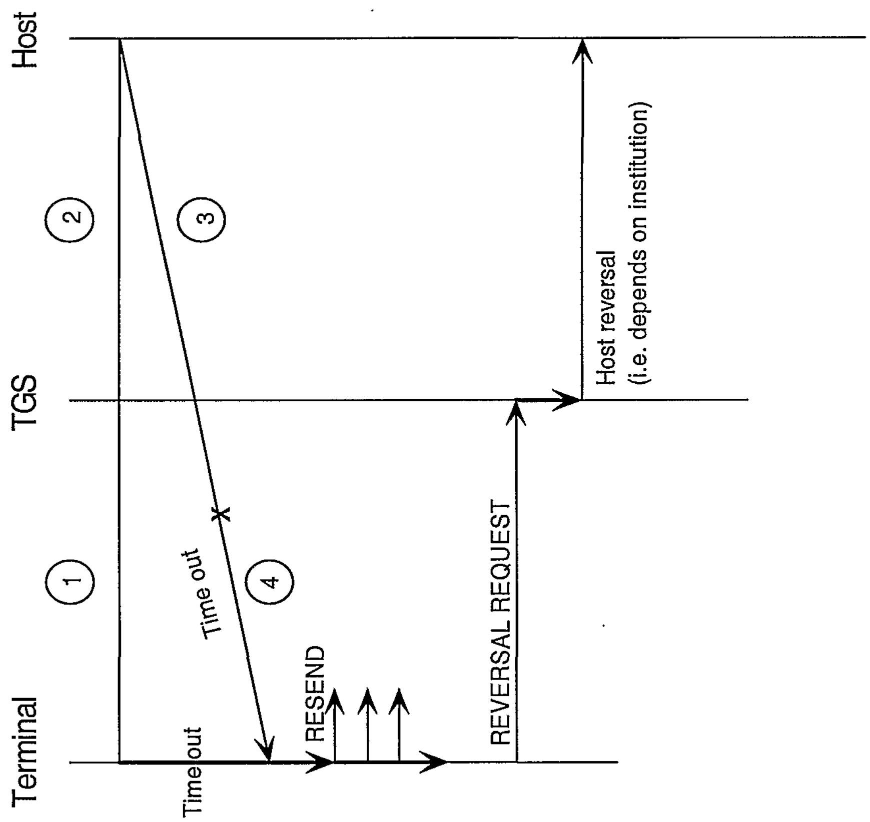

Host
TGS
Terminal
1
2
3
4
Time out
Time out
RESEND
REVERSAL REQUEST
Host reversal
(i.e. depends on institution)

Figure 5

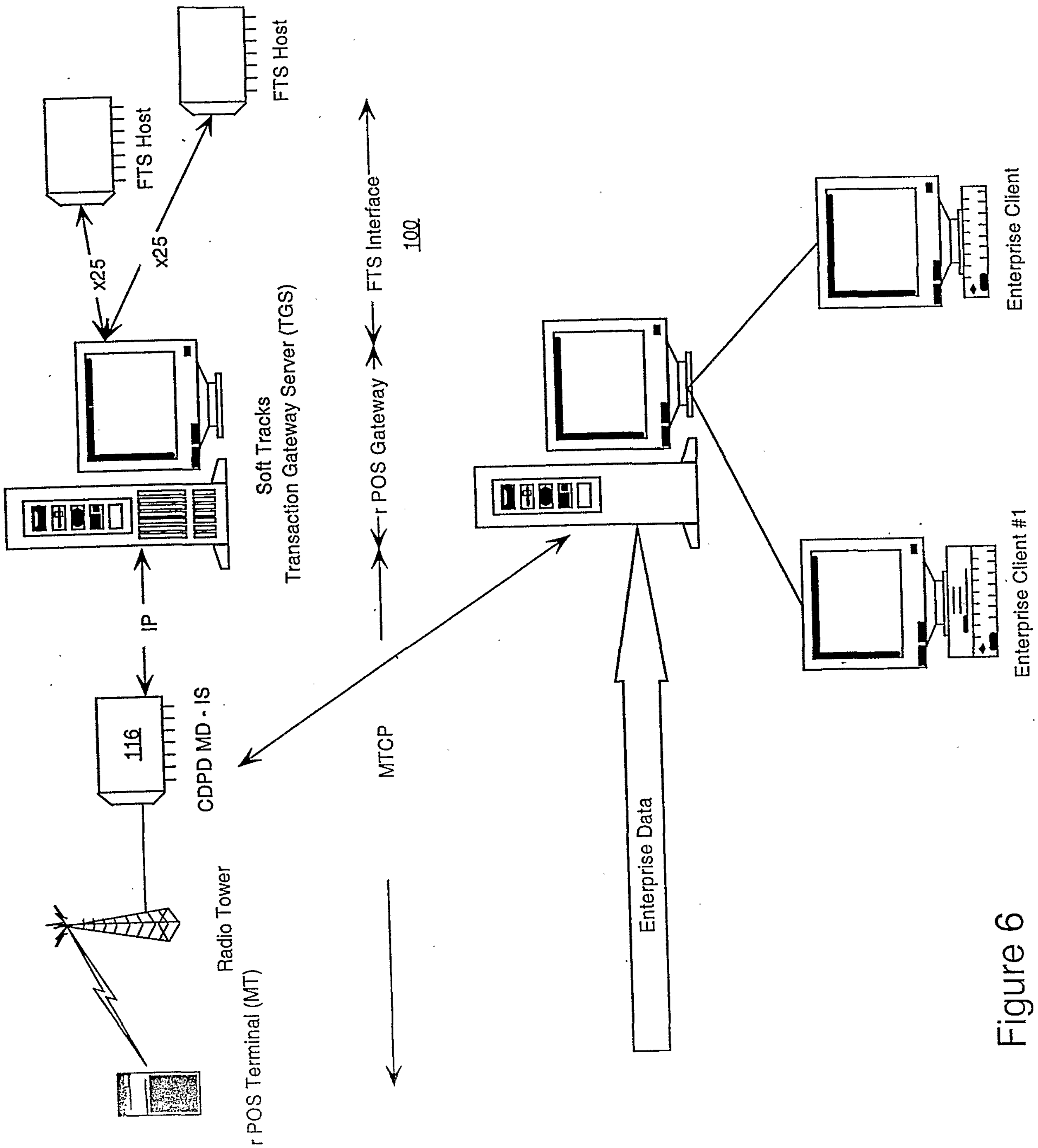

FTS Host
FTS Host
x25
x25
Soft Tracks
Transaction Gateway Server (TGS)
FTS Interface
100
r POS Gateway
IP
116
CDPD MD - IS
MTCP
Radio Tower
r POS Terminal (MT)
Enterprise Data
Enterprise Client
Enterprise Client #1

Figure 6